What is Claimed is:

1.　A computer readable data storage medium comprising a software class for supporting a plurality of servers in an intelligent messaging network, the class comprising:

　　　a first code segment handling registration of the servers with the intelligent messaging network;

　　　a second code segment for connecting the servers to one another; and

　　　a third code segment encapsulating communication between servers.

2.　The computer readable data storage medium of claim 1, wherein the first code segment specifies a server class and a server type.

3.　The computer readable data storage medium of claim 1, wherein the first code segment further specifies at least one of a packet header, an IP address, and a listener port.

4.　The computer readable data storage medium of claim 1, wherein the third code segment generates a standard packet for communication between the servers.

5.　The computer readable data storage medium of claim 4, wherein the standard packet includes at least one of:

　　　a header length;

　　　protocol flags;

　　　packet length;

　　　database ID;

　　　link station ID;

　　　message ID;

　　　customer ID;

　　　port number;

　　　network header; and

　　　message body.

6.   The computer readable data storage medium of claim 5, wherein the network header includes at least one of:

a compression indicator;

a security indicator;

a service type indicator;

a message type indicator; and

a server ID.

7.   The computer readable data storage medium of claim 1, further comprising:

a fourth code segment encapsulating a transport header;

a fifth code segment notifying a sender of a success or failure of a transmission;

a sixth code segment segmenting messages over a pre-determined length into message segments;

a seventh code segment assembling messages segments into messages;

an eighth code segment resending messages that are not acknowledged within a pre-determined time;

a ninth code segment pacing a transmission of messages larger than a pre-determined number of segments;

a tenth code segment detecting duplicate message segments; and

an eleventh code segment detecting duplicate messages.

8.   The computer readable data storage medium of claim 1, further comprising:

a fourth code segment generating acknowledgement messages;

a fifth code segment processing the acknowledgement messages;

a sixth code segment compressing and decompressing messages; and

a seventh code segment encrypting and decrypting messages.

9.   The computer readable data storage medium of claim 7, further comprising:

a twelfth code segment encapsulating a communication layer.

10. The computer readable data storage medium of claim 8, further comprising:

an eighth code segment processing application specific messages; and

a ninth code segment providing special compression services;

a tenth code segment providing special security services.

11. A computer readable storage medium comprising a software class providing graphical user interfaces, the class comprising:

a first code segment providing a base registry key for storage of server settings and an user interface for viewing or editing the server settings;

a second code segment providing screen based error logging; and

a third code segment for writing system errors to an event log.

12. The computer readable data storage medium of claim 11, wherein the third code segment specifies a batch file to be executed when a specified error occurs.

13. The computer readable data storage medium of claim 11, further comprising,

a fourth code segment providing a user interface for transport settings.

a fifth code segment logging each inbound and outbound message; and

a sixth code segment displaying pre-selected statistics.

14. The computer readable data storage medium of claim 13, further comprising a seventh code segment providing a separate logging interface for logging application errors.

15. The computer readable data storage medium of claim 11, wherein the transport settings include at least one of: a maximum number of retries, a retry timeout interval, and a segment size.

16. The computer readable data storage medium of claim 11 wherein the pre-selected statistics include at least one of a number of messages sent/received and a number of ACK/NACK sent/received.

17.   The computer readable data storage medium of claim 11, further comprising:

a fourth code segement providing a GUI for displaying a log of inbound/outbound messages;

a fifth code segment logging each inbound and outbound message; and

a sixth code segment displaying pre-selected statistics.

18.   A software development system for developing client applications, comprising:

a utility library;

a security library; and

a transport library, wherein the transport library is independent from both the security library and the utility library.

19.   The system of claim 18, wherein the utility library comprises:

a first code segment for handling streaming input/output messages to the client application;

a second code segment providing compression services for messages;

a third code segment creating a message header for messages generated by the client application; and

a fourth code segment building authentication messages.

20.   The system of claim 19, wherein the second code segment determines if a message is to be encoded and provides encoding services.

21.   The system of claim 19, wherein the fourth code segment determines an authentication status.

22.   The system of claim 18, wherein the transport library comprises:

a first code segment for specifying a target of a message generated by the client application;

a second code segment notifying a sender of a success or failure of a transmission;

a third code segment segmenting messages over a pre-determined length into message segments;

a fourth code segment assembling messages segments into messages;

an fifth code segment for resending messages that are not acknowledged within a pre-determined time;

a sixth code segment handling duplicate message segments; and

an seventh code segment handling duplicate messages.

23.   The system of claim 18, wherein the security library comprises:

a first code segment establishing a secret key between the client application and a server;

a second code segment for encrypting messages; and

a third code segment for decrypting messages.

24.   A method for supporting a plurality of servers in an intelligent messaging network, comprising:

providing registration of the servers with the intelligent messaging network;

providing connectivity of the servers to one another; and

encapsulating communication between servers.

25.   The computer readable data storage medium of claim 24, further comprising specifing a server class and a server type during registration.

26.   The computer readable data storage medium of claim 24, further comprising specifing at least one of a packet header, an IP address and a listener port during registration.

27.   The computer readable data storage medium of claim 24, further comprising generating a standard packet for communication between the servers during encapsulation.

28.   The computer readable data storage medium of claim 27, wherein the standard packet includes at least one of:
        a header length;
        protocol flags;
        packet length;
        database ID;
        link station ID;
        message ID;
        customer ID;
        port number;
        network header; and
        message body.

29.   The computer readable data storage medium of claim 28, wherein the network header includes at least one of:
        a compression indicator;
        a security indicator;
        a service type indicator;
        a message type indicator; and
        a server ID.

30. The computer readable data storage medium of claim 24, further comprising:

encapsulating a transport header;

notifying a sender of a success or failure of a transmission;

segmenting messages over a pre-determined length into message segments;

assembling the messages segments into messages;

resending messages that are not acknowledged within a pre-determined time;

pacing a transmission of messages larger than a pre-determined number of segments;

detecting duplicate message segments; and

detecting duplicate messages.

31. The computer readable data storage medium of claim 24, further comprising:

generating acknowledgement messages;

processing the acknowledgement messages;

compressing and decompressing messages; and

encrypting and decrypting messages.

32. The computer readable data storage medium of claim 30, further comprising:

encapsulating a communication layer.

33. The computer readable data storage medium of claim 31, further comprising:

processing application specific messages;

providing special compression services;

providing special security services.

34. A computer readable storage medium comprising a software class providing graphical user interfaces, the class comprising:

providing a base registry key for storage of server settings and an user interface for viewing or editing the server settings;

providing screen based error logging; and

writing system errors to an event log.

35. The computer readable data storage medium of claim 34, wherein the writing step specifies a batch file to be executed when a specified error occurs.

36. The computer readable data storage medium of claim 34, further comprising,

providing a user interface for transport settings.

logging each inbound and outbound message; and

displaying pre-selected statistics.

37. The computer readable data storage medium of claim 36, further comprising:

providing a separate logging interface for logging application errors.

38. The computer readable data storage medium of claim 34, wherein the transport settings include at least one of: a maximum number of retries, a retry timeout interval, and a segment size.

39. The computer readable data storage medium of claim 34 wherein the pre-selected statistics include at least one of a number of messages sent/received and a number of ACK/NACK sent/received.

40. The computer readable data storage medium of claim 34, further comprising:

providing a GUI for displaying a log of inbound/outbound messages;

logging each inbound and outbound message; and

displaying pre-selected statistics.

41. A method for developing client applications, comprising:

providing utility components for the client application to run;

providing security components for encryption; and

providing transport components for data communication over wireless networks, wherein the transport components are independent from both the security components and the utility components.

42. The system of claim 41, wherein the utility components include elements:

handling streaming input/output messages to the client application;

providing compression services for messages;

creating a message header for messages generated by the client application; and

building authentication messages.

43. The system of claim 42, wherein the compression services determine if a message is to be encoded and provides encoding services.

44. The system of claim 42, further comprising determining an authentication status.

45. The system of claim 41, wherein the transport components comprise elements:

specifying a target of a message generated by the client application;

notifying a sender of a success or failure of a transmission;

segmenting messages over a pre-determined length into message segments;

assembling message segments into messages;

resending messages that are not acknowledged within a pre-determined time;

handling duplicate message segments; and

handling duplicate messages.

46.   The system of claim 41, wherein the security components comprise elements:

establishing a secret key between the client application and a server;

encrypting messages; and

decrypting messages.

47.   An SDK, comprising:

registration components for handling registration of servers with an intelligent messaging network;

connectivity components for connecting the servers to one another; and

communication components for encapsulating communication between the servers.

48.   A SDK, comprising:

a base registry key for storage of server settings and an user interface for viewing or editing the server settings;

screen based error logging; and

error components for writing system errors to an event log.

49. An SDK, comprising:

utility components providing functions for a client application to run;

security components for encryption of messages; and

transport components for data communication over wireless networks, wherein the transport components are independent from both the security components and the utility components.